

## Pulp - Issue #6346

### Remote fields username and password not showing up in REST docs

03/16/2020 07:00 PM - daviddavis

<b>Status:</b>	CLOSED - CURRENTRELEASE	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	fao89	<b>Estimated time:</b>	0:00 hour
<b>Category:</b>		<b>Groomed:</b>	No
<b>Sprint/Milestone:</b>	3.4.0	<b>Sprint Candidate:</b>	No
<b>Severity:</b>	2. Medium	<b>Tags:</b>	
<b>Version:</b>		<b>Sprint:</b>	Sprint 72
<b>Platform Release:</b>		<b>Quarter:</b>	
<b>OS:</b>			
<b>Triaged:</b>	Yes		
<b>Description</b>			
Remote serializer has fields username and password:  <a href="https://github.com/pulp/pulpcore/blob/c652e5e3c3bd99e60896463ea01c35e363031e15/pulpcore/app/serializers/repository.py#L86-L97">https://github.com/pulp/pulpcore/blob/c652e5e3c3bd99e60896463ea01c35e363031e15/pulpcore/app/serializers/repository.py#L86-L97</a>  These fields don't seem to show up in the REST api docs:  <a href="https://docs.pulpproject.org/en/master/nightly/restapi.html#operation/remotes_file_file_create">https://docs.pulpproject.org/en/master/nightly/restapi.html#operation/remotes_file_file_create</a>			
<b>Related issues:</b>			
Blocked by Pulp - Task #6421: Review of write_only fields in pulp and plugins		<b>CLOSED - COMPLETE</b>	

#### Associated revisions

##### Revision 5688324f - 05/11/2020 04:17 PM - Fabricio Aguiar

Return username & password when requesting Remote

<https://pulp.plan.io/issues/6346> closes #6346

#### History

##### #1 - 03/17/2020 03:32 PM - fao89

- Triaged changed from No to Yes
- Sprint set to Sprint 68

##### #2 - 03/19/2020 04:30 PM - fao89

- Status changed from NEW to ASSIGNED
- Assignee set to fao89

##### #3 - 03/19/2020 07:40 PM - fao89

They do not appear because they are write\_only: <https://github.com/pulp/pulpcore/pull/361/files> it is a drf\_yasg limitation: <https://github.com/axnsan12/drf-yasg/issues/70#issuecomment-368233077> For fixing this, we would have to apply this patch: <https://github.com/axnsan12/drf-yasg/issues/70#issuecomment-485050813> Which would change our bindings

##### #4 - 03/20/2020 04:21 PM - rchan

- Sprint changed from Sprint 68 to Sprint 69

#### #5 - 03/20/2020 10:21 PM - fao89

- Status changed from ASSIGNED to POST

<https://github.com/pulp/pulpcore/pull/600>

#### #6 - 03/25/2020 02:15 PM - pulpbot

PR: <https://github.com/pulp/pulpcore/pull/600>

#### #7 - 03/25/2020 02:58 PM - daviddavis

After a discussion on IRC with @dkliban, [fao89](#), and [bmbouter](#), we decided that pulp should use SecretCharFields for write-only data and to either document (or validate) that serializers shouldn't use the write\_only option on fields.

#### #8 - 03/25/2020 03:12 PM - bmbouter

I'm +1 on using SecretCharField. What would happen with a bindings user:

1. Use the bindings to read() the an endpoint whose serializer is using SecretCharField, say as field password
2. Call an update() on the bindings object to update another attribute asdf

Is there any possibility that while setting asdf the value of password is also changed? Or does it require an explicit update of the password field itself? What about PUT versus PATCH calls?

#### #9 - 03/25/2020 06:16 PM - dkliban@redhat.com

It looks like if you take the FileRemote object returned by the read() or create() method and pass it to the update() method, the client\_cert field gets updated to a new value. This will be true for any SecretCharField.

```
(Pdb) file_remote
{'ca_cert': None,
 'client_cert': 'b3226466e6d9c43c7058f69e1ff41daaf688cd223c084faa3e28202813ecff28',
 'client_key': None,
 'download_concurrency': 20,
 'name': 'bar25',
 'policy': 'immediate',
 'proxy_url': None,
 'pulp_created': datetime.datetime(2020, 3, 25, 17, 9, 13, 631685, tzinfo=tzlocal()),
 'pulp_href': '/pulp/api/v3/remotes/file/file/e42eea0d-e583-45ce-bdca-342169385cb2/',
 'pulp_last_updated': datetime.datetime(2020, 3, 25, 17, 9, 13, 631703, tzinfo=tzlocal()),
 'tls_validation': True,
 'url': 'https://repos.fedorapeople.org/pulp/pulp/demo_repos/test_file_repo/PULP_MANIFEST'}
(Pdb) fileremotes.update(file_remote.pulp_href, file_remote)
{'task': '/pulp/api/v3/tasks/ec6a2bb6-4ce4-4a0e-ab1c-62d37aecfd27/'}
(Pdb) file_updated_remote = fileremotes.read(file_remote.pulp_href)
(Pdb) file_updated_remote
{'ca_cert': None,
 'client_cert': '7fc2a6b69d81c4581eac98454217b173a8b23256eefad0bb9eaabe199d8baae8',
 'client_key': None,
 'download_concurrency': 20,
 'name': 'bar25',
 'policy': 'immediate',
 'proxy_url': None,
 'pulp_created': datetime.datetime(2020, 3, 25, 17, 9, 13, 631685, tzinfo=tzlocal()),
 'pulp_href': '/pulp/api/v3/remotes/file/file/e42eea0d-e583-45ce-bdca-342169385cb2/',
 'pulp_last_updated': datetime.datetime(2020, 3, 25, 17, 10, 37, 747011, tzinfo=tzlocal()),
 'tls_validation': True,
 'url': 'https://repos.fedorapeople.org/pulp/pulp/demo_repos/test_file_repo/PULP_MANIFEST'}
```

write\_only=True

pulp-2to3-migration:

```
validate = serializers.BooleanField(
    help_text=_('If `True`, migration cannot happen without successful validation '
                'of the Migration Plan'),
    required=False,
    default=False,
    write_only=True
)
dry_run = serializers.BooleanField(
    help_text=_('If `True`, performs validation of a Migration Plan only, no migration is '
                'run.'),
    required=False,
    default=False,
    write_only=True
)
```

pulp-certguard:

```
ca_certificate = serializers.FileField(
    help_text=_('The Certificate Authority certificate.'),
    write_only=True
)
```

pulp\_container:

```
content_units = serializers.ListField(
    help_text=_('A list of content units to operate on.'),
    write_only=True,
    required=False
)
source_repository = serializers.HyperlinkedRelatedField(
    help_text=_('A URI of the repository to copy content from.'),
    queryset=models.ContainerRepository.objects.all(),
    view_name='repositories-container/container-detail',
    label=_('Repository'),
    write_only=True,
    required=False,
)
source_repository_version = NestedRelatedField(
    help_text=_('A URI of the repository version to copy content from.'),
    view_name='versions-detail',
    lookup_field='number',
    parent_lookup_kwargs={'repository_pk': 'repository__pk'},
    queryset=RepositoryVersion.objects.all(),
    write_only=True,
    required=False,
)
containerfile = serializers.FileField(
    help_text=(
        "An uploaded Containerfile that should be used to run buildah."
    ),
    required=False,
    write_only=True,
)
```

pulpcore:

```
relative_path = serializers.CharField(
    help_text=_('Path where the artifact is located relative to distributions base_path'),
```

```

        validators=[fields.relative_path_validator],
        write_only=True,
    )
    username = serializers.CharField(
        help_text='The username to be used for authentication when syncing.',
        write_only=True,
        required=False,
        allow_null=True,
    )
    password = serializers.CharField(
        help_text='The password to be used for authentication when syncing.',
        write_only=True,
        required=False,
        allow_null=True,
    )
    add_content_units = serializers.ListField(
        help_text=_('A list of content units to add to a new repository version. This content is '
                    'added after remove_content_units are removed.'),
        write_only=True,
        required=False
    )
    remove_content_units = serializers.ListField(
        help_text=_('A list of content units to remove from the latest repository version. "
                    "You may also specify '*' as an entry to remove all content. This content is "
                    "removed before add_content_units are added.'),
        write_only=True,
        required=False
    )
    file = serializers.FileField(
        help_text=_("A chunk of the uploaded file."),
        write_only=True,
    )

    sha256 = serializers.CharField(
        help_text=_("The SHA-256 checksum of the chunk if available."),
        required=False,
        allow_null=True,
        write_only=True,
    )
    file = FileField(
        help_text=(
            "An uploaded file that should be turned into the artifact of the content unit."
        ),
        required=False,
        write_only=True,
    )
    repository = DetailRelatedField(
        help_text=(
            "A URI of a repository the new content unit should be associated with."
        ),
        required=False,
        write_only=True,
        queryset=Repository.objects.all(),
    )

```

**#11 - 03/26/2020 03:20 PM - bmbouter**

I see the write\_only usage falling into two categories.

### **write\_only for secrecy**

These are fields that are unsafe to be re-shown to the user. These are the ones that should have a solution similar to SecretCharField that shows you the sha256 when viewed, and refuses to overwrite data if you're submitting the sha256 of the data already stored.

The only ones I think that fall in this category are: Remote.username, Remote.password

### **write\_only for correctness**

These are all the others, and I think these should be viewable but only allowed to set once. These would be all other fields.

**#12 - 03/26/2020 03:30 PM - daviddavis**

I think there's also a third category: a field that is not actually stored in the database. The fields on MigrationPlanRunSerializer[0] are an example.

In such cases, perhaps we just roll a write only serializer to be used for POST/PUT/PATCH (but not GET) and not set the write\_only option on these fields.

[0] <https://git.io/Jv9EU>

**#13 - 03/26/2020 03:37 PM - bmbouter**

daviddavis wrote:

I think there's also a third category: a field that is not actually stored in the database. The fields on MigrationPlanRunSerializer[0] are an example.

In such cases, perhaps we just roll a write only serializer to be used for POST/PUT/PATCH (but not GET) and not set the write\_only option on these fields.

[0] <https://git.io/Jv9EU>

Agreed, sounds great.

**#14 - 03/26/2020 04:13 PM - pulpbot**

PR: <https://github.com/pulp/pulpcore/pull/611>

**#15 - 03/26/2020 06:48 PM - pulpbot**

PR: <https://github.com/pulp/pulpcore/pull/612>

**#16 - 03/26/2020 07:23 PM - pulpbot**

PR: <https://github.com/pulp/pulpcore/pull/613>

**#17 - 03/26/2020 09:48 PM - fao89**

how about the initial issue that made me hide write\_only fields? <https://pulp.plan.io/issues/5622>

**#18 - 03/27/2020 06:57 PM - fao89**

Two options:

1 - <https://github.com/pulp/pulpcore/pull/600>

- write\_only fields do not show up in the docs
- username and password are not write\_only anymore
- browsable API response == httpie response == Rest API docs response

2- <https://github.com/pulp/pulpcore/pull/613>

- write\_only fields show up in the docs
- (browsable API response == httpie response) != Rest API docs response

**#19 - 03/31/2020 07:28 PM - daviddavis**

- Blocked by Task #6421: Review of write\_only fields in pulp and plugins added

**#20 - 04/03/2020 06:13 PM - rchan**

- Sprint changed from Sprint 69 to Sprint 70

**#21 - 04/20/2020 02:37 PM - rchan**

- Sprint changed from Sprint 70 to Sprint 71

**#22 - 05/01/2020 06:03 PM - rchan**

- Sprint changed from Sprint 71 to Sprint 72

**#23 - 05/08/2020 10:20 PM - pulpbot**

PR: <https://github.com/pulp/pulpcore/pull/695>

**#24 - 05/11/2020 09:06 PM - Anonymous**

- Status changed from POST to MODIFIED

Applied in changeset [pulpcore|5688324ff35fb67776f7f0630275cfbfc84ea163](https://pulp.plan.io/changeset/pulpcore|5688324ff35fb67776f7f0630275cfbfc84ea163).

**#25 - 05/15/2020 07:40 AM - Anonymous**

Looks like a fix has landed in master, could we get it pushed to stable please?

**#26 - 05/18/2020 02:45 PM - daviddavis**

@snecklifter, we have a 3.4 release planned for next week. Would that work for you?

**#27 - 05/19/2020 08:34 PM - pulpbot**

PR: <https://github.com/pulp/pulpcore/pull/707>

**#28 - 05/20/2020 01:40 PM - Anonymous**

[daviddavis](#) yes, thanks

**#29 - 05/27/2020 09:28 PM - dkliban@redhat.com**

- *Sprint/Milestone set to 3.4.0*

**#30 - 05/28/2020 07:00 PM - dkliban@redhat.com**

- *Status changed from MODIFIED to CLOSED - CURRENTRELEASE*